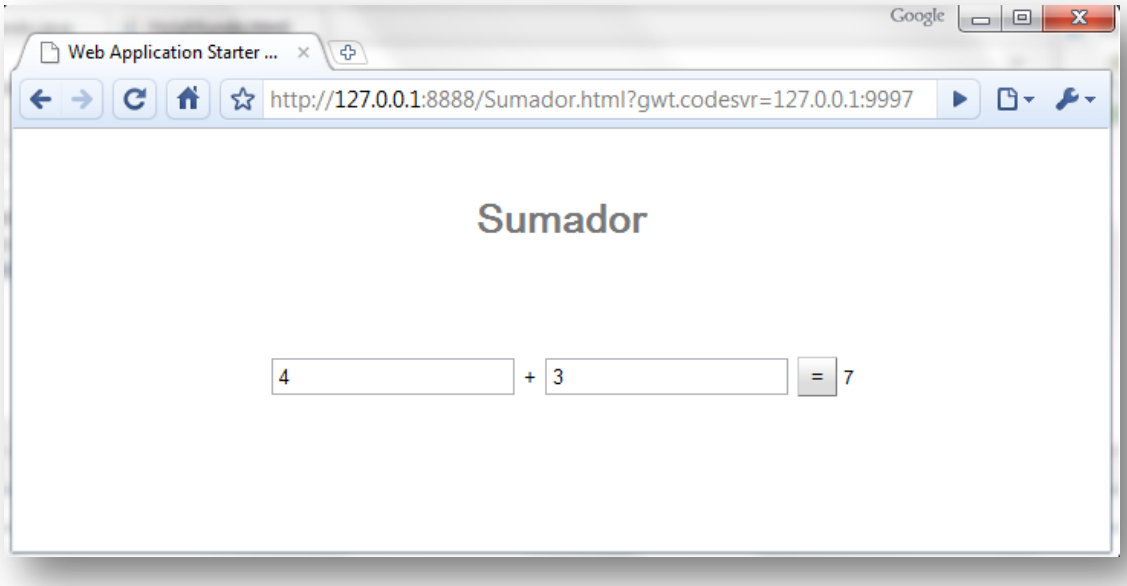
HANDS ON GWT

Actividad II

Objetivo:

Crear una aplicación similar a la desarrollada en la actividad 01, que muestre dos cajas de texto sobre las que se ingresen dos números que serán sumados luego de hacer click sobre un botón. Esta vez la suma no se realizará en el cliente, si no que se utilizará un **servicio de suma remota**, que se ejecutará en el servidor. El resultado se mostrará sobre un componente Label como en la actividad 01.



Fecha de Entrega:

El trabajo se admitirá hasta las 0 Horas del día 16 de Junio del 2010.

Formato de Entrega:

Se deberá entregar el archivo *Sumador.java* correspondiente al EntryPoint de la aplicación, el archivo *Sumador.html* correspondiente a la Host Page y el archivo de configuración *web.xml*. También deberán entregarse los archivos de las clases de soporte al servicio de suma remota, que serán *Suma.java, SumadorService.java, SumadorServiceAsync.java, SumadorServletImpl.java y NumeroNegativoException.java.*

Desarrollo:

Crear un proyecto GWT con nombre “SumadorRemoto” por medio del asistente de proyecto del Google Eclipse Plugin. El paquete raíz del proyecto debe ser `com.logikas.handsongwt.actividad02`. Los paquetes **shared** y **server** serán conservados y contendrán las clases de soporte al servicio de suma remota.

El servicio de suma remota será provisto por la interface de servicio **SumaService**, que será definida en el paquete adecuado (**client**, **shared** o **server**).

```
public interface SumaService extends RemoteService {
      int sumar(Suma suma) throws NumeroNegativoException;
}
```

La interface **SumaService** será acompañada por correspondiente implementación e interface asíncrona, que también deberán ser definidas en los paquetes adecuados (**client**, **shared** o **server**).

La clase Suma, será un objeto de intercambio de datos, que deberá ser definido en un paquete adecuado, y contendrá las propiedades numero1 y numero2, según el siguiente ***pseudocódigo***:

```
class Suma {
   int getNumero1();
   int getNumero2();
}
```

El **servicio de suma remota** deberá comprobar que ambos números sean mayores que cero. En el caso de que esta condición no se cumpla, deberá lanzar la excepción **NumeroNegativoException**.

```
package com.logikas.handsongwt.actividad02.shared;

public class NumeroNegativoException extends Exception {

  public NumeroNegativoException() {
  }

  public NumeroNegativoException(String message) {
    super(message);
  }

  public NumeroNegativoException(Throwable caught) {
    super(caught);
  }

  public NumeroNegativoException(String message, Throwable caught) {
    super(message, caught);
  }
}
```

El desarrollo de la interfaz visual será similar al desarrollado en la actividad 01, pero esta vez, cuando se haga click sobre el botón “sumar”, se invocará el servicio `sumar()` definido en la interfaz **SumaService**. Si el servicio remoto dispara una excepción **NumeroNegativoException**, se mostrará el mensaje “error de numero negativo” en el campo “resultado”. Si el servicio dispara cualquier otro tipo de excepción, se mostrará el mensaje “error en la red” en el campo “resultado”.

Luego de hacer click sobre el botón “sumar”, mientras el mecanismo de RPC se encuentre intercambiando datos con el servidor, se deberán deshabilitar las cajas de texto “numero1” y “numero2” y el botón “sumar”, para evitar que el usuario cargue datos mientras se realiza el procesamiento. Para deshabilitar un widget se debe establecer el valor **false** en la propiedad booleana “enabled”.

```
// deshabilitar el widget
widget.setEnabled(false);
```

Cuando finalice la comunicación el servidor, tanto en caso de éxito como de error, se volverán a habilitar los widgets anteriormente deshabilitados.

```
// habilitar el widget
widget.setEnabled(true);
```